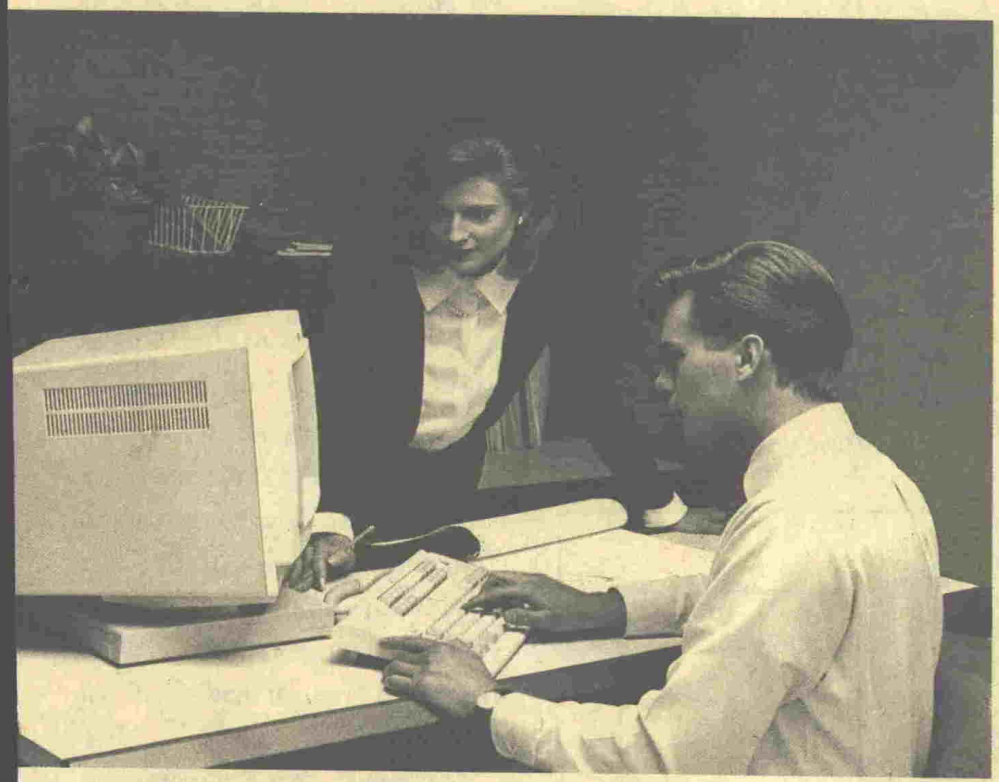
การเก็บข้อมูลสเปรดชีตของโลตัส 1-2-3 เป็นอย่างไร



ยืน ภู่วรวรรณ

สเปรดชีตเป็นซอฟต์แวร์สำเร็จรูปที่ผู้ใช้ไมโครคอมพิวเตอร์รู้จักกันเป็นอย่างดี สเปรดชีตเป็นโปรแกรมอเนกประสงค์ที่ทำให้ผู้ใช้ทำงานในสิ่งต่าง ๆ ได้อย่างกว้างขวาง การพัฒนาโปรแกรมสเปรดชีตก็ยังคงดำเนินต่อไปและมีแนวโน้มว่าจะมีการเพิ่มขีดความสามารถได้อีกมาก อย่างไรก็ดีโครงสร้างข้อมูลของสเปรดชีตเป็นเรื่องที่ต้องอาศัยเทคนิคและหลักการบางอย่าง เพื่อให้ได้ประสิทธิภาพการทำงานอย่างเต็มที่ ในบทความนี้จะนำเอาโครงสร้างข้อมูลของโลตัส ซึ่งเป็นโปรแกรมสเปรดชีตที่มีผู้ใช้งานมากที่สุดมาเสนอ ซึ่งจะทำให้เกิดความเข้าใจและหาทางนำไปใช้ประโยชน์ต่อไปได้

ความเป็นมา

หากจะย้อนไปในอดีตตั้งแต่ครั้งการเริ่มต้นมีการใช้สเปรดชีตในปี ค.ศ. 1979 ซึ่งถือได้ว่าเป็นจุดเริ่มต้นของโปรแกรมสเปรดชีต โปรแกรมสเปรดชีตตัวแรกคือวิสิแคลค์ การพัฒนาวิสิแคลค์ทำให้สเปรดชีตเป็นที่รู้จักกันและนิยมชมชอบของผู้ใช้ไมโครคอมพิวเตอร์เป็นอย่างยิ่ง วิสิแคลค์ขายดีและจัดได้ว่าเป็นโปรแกรมทอปฮิตขณะนั้น และต่อมาก็มีซูเปอร์แคลค์และโลตัส 1-2-3 ซึ่งก็ยังมีสเปรดชีตอีกหลาย ๆ โปรแกรมติดตามมา

โครงสร้างของวิสิแคลค์

วิสิแคลค์เก็บข้อมูลอย่างเรียบง่าย โดยเก็บข้อมูลเป็นไฟล์ที่มีข้อมูลเป็นรหัส ASCII เช่น ข้อมูลในวิสิแคลค์เป็นดังนี้

	A	B
1	Sales	6148
2	Cost	5000
3	Profit	+B1-B2

เมื่อวิสิแคลค์เก็บข้อมูลก็จะเก็บข้อมูลเป็นเท็กซ์ไฟล์ซึ่งถ้าพิมพ์ข้อมูลออกมาดูจะพบข้อมูล ดังนี้

```
> A1:"Sales
> A2:"Cost
> A3:"Profit
> B1:6148
> B2:5000
> B3: / F$B1-B2
/W1
```

/ GFR
/ GC9
/ X>A1:>A1:

การเก็บข้อมูลแบบเท็กซ์นี้เองทำให้วิสิแคลค์ต้องเสียพื้นที่ในหน่วยความจำสูงขึ้น เช่นเมื่อเก็บเลขจำนวนเต็มก็จะต้องใช้หลายไบต์ในการเก็บข้อมูลและการแปลผลข้อมูลเพื่อใช้ในการประมวลผลก็ทำได้ช้า จึงทำให้การทำงานของวิสิแคลค์ช้า จึงได้มีผู้พยายามคิดโครงสร้างข้อมูลแบบต่าง ๆ เพื่อให้ทำงานได้มีประสิทธิภาพดียิ่งขึ้น การเก็บข้อมูลจึงใช้รหัสเข้ามาช่วย

ลักษณะของข้อมูลที่เก็บในดิสค์จะมีโครงสร้างเสมือนเป็นเมตาไฟล์ (Metafile) คือเก็บข้อมูลในรูปรหัสบางส่วนเพื่อให้มีรายละเอียดของสเปรดชีตทุกอย่างอยู่ครบ แต่เก็บข้อมูลด้วยหลักของการใช้ที่น้อยทำงานได้รวดเร็ว

โครงสร้างของข้อมูลโลตัสในดิสค์

โปรแกรมสเปรดชีตโลตัส 1-2-3 ได้ออกแบบมาสำหรับเครื่อง 16 บิต ที่มีขีดความสามารถสูงกว่าวิสิแคลค์ที่วิ่งอยู่บนเครื่องแอปเปิ้ลทู โครงสร้างของข้อมูลของโลตัสได้เปลี่ยนแปลงไปจากวิสิแคลค์มาก การเก็บข้อมูลตัวเลขใช้ตัวเลขไบนารีทำให้ลดขนาดจำนวนไบต์ลงไปได้มาก ตัวเลขขนาด-21876 จะเก็บเพียง 2 ไบต์เท่านั้น (ในวิสิแคลค์เก็บ 6 ไบต์) การลดขนาดข้อมูลลงทำให้การอ่านข้อมูลจากดิสค์ทำได้เร็วกว่า อย่างไรก็ตามโปรแกรมจะต้องทำการแปลข้อมูลเพื่อนำมาใช้งานในหน่วยความจำ การเก็บข้อมูลในดิสค์ของโลตัสจึงมีรูปแบบที่ออกแบบมาโดยเฉพาะ แฟ้มข้อมูลของโลตัสจึงแปลงไปใช้ในโปรแกรมอื่นเช่นดีเบสไม่ได้โดยตรง จำเป็นจะต้องมีโปรแกรมฟอร์แมตข้อมูลให้

โครงสร้างข้อมูลภายในยังเป็นเรคอร์ด

ข้อมูลเวอร์กชีตของโลตัส 1-2-3 เก็บในไฟล์ในฟอร์แมตที่เป็นเรคอร์ด โดยมีรูปแบบของฟอร์แมตหลายแบบแต่ละแบบก็จะมีความหมายเฉพาะการที่โลตัสสร้างฟอร์แมตไฟล์ของตนเองก็เพื่อให้การทำงานของโปรแกรมมีความรวดเร็วยิ่งขึ้น อย่างไรก็ตาม เมื่อโลตัส 1-2-3 เก็บข้อมูลกราฟ ผู้ใช้จะต้องเก็บไว้ในอีกฟอร์แมตหนึ่งที่มีลักษณะแตกต่างออกไปจากเดิมโดยสิ้นเชิง โครงสร้างแฟ้มข้อมูลของโลตัส 1-2-3 ได้รับการปรับปรุงและนำไปใช้ในโปรแกรมซิมโฟนีด้วยเช่นกัน

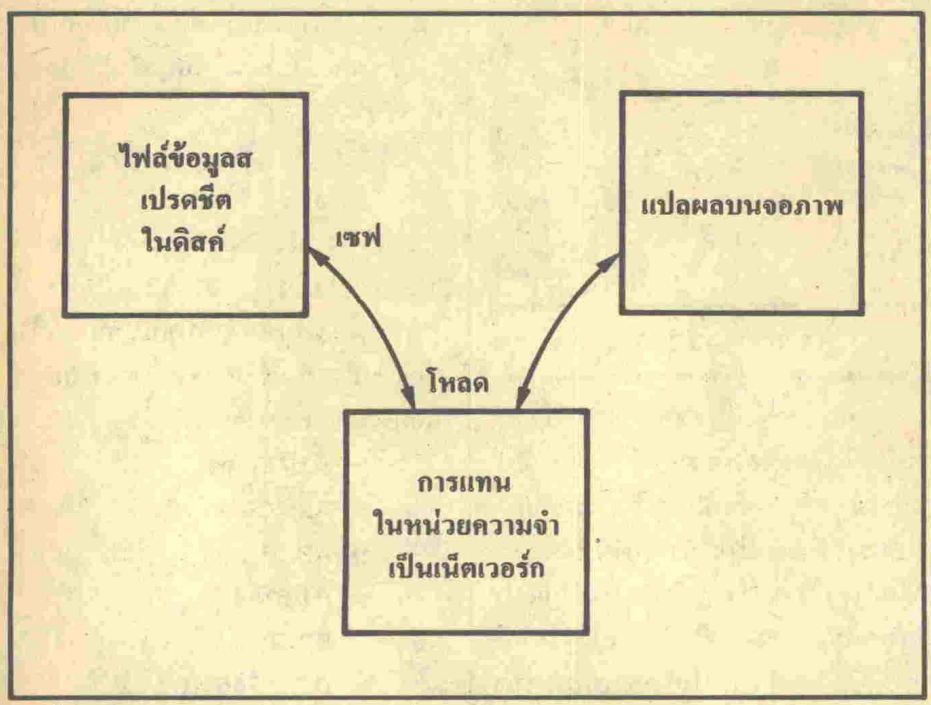
โครงสร้างพื้นฐานของข้อมูลโลตัส

ทุก ๆ เรคอร์ดของโลตัสจะเก็บในลักษณะฟอร์แมตที่คล้ายกัน อย่างไรก็ดีแต่ละเรคอร์ดอาจมีความยาวไม่เท่ากัน โครงสร้างของเรคอร์ดแบ่งออกเป็นสองส่วน คือ ส่วนของเฮดเดอร์ของเรคอร์ด และส่วนของข้อมูล โดยโครงสร้างเขียนได้ดังรูปที่ 2

เฮดเดอร์ของเรคอร์ดประกอบด้วยรหัสของเรคอร์ดและข้อมูลที่บอกความยาวของเรคอร์ดนั้น ๆ ส่วนของข้อมูลก็คือที่เก็บข้อมูลซึ่งอาจจะมีหลายฟิลด์ประกอบกันอยู่ โครงสร้างของแต่ละเรคอร์ดจะได้รับการตีความหมายได้ว่าเรคอร์ดนั้น ๆ มีข้อมูลอะไรอยู่ การแปลความหมายนี้จะกระทำโดยโลตัส 1-2-3 อย่างถูกต้องและเก็บโครงสร้างลงในหน่วยความจำ

รหัสของเรคอร์ดเป็นตัวเลขขนาด 2 หลัก ที่จะเป็นตัวบอกว่าเรคอร์ดนี้คืออะไร นอกจากนี้รหัสที่ใช้ยังให้ความหมายถึงพารามิเตอร์ของเวอร์กชีตที่ได้รับการเซตไว้ในคำสั่ง เช่น รูปแบบของการให้การคำนวณแบบอัตโนมัติ การกำหนดฟอร์แมตของเวอร์กชีต

ส่วนที่บอกความยาวของข้อมูลเป็นตัวเลขขนาด 2 ไบต์ ซึ่งเก็บข้อมูลบอกความยาวเป็นไบต์ของเรคอร์ด การมีตัวบอกความยาวก็เพราะว่าทุกเรคอร์ด



รูปที่ 1 โครงสร้างข้อมูลของสเปรดชีต

มีความยาวไม่เท่ากัน ดังนั้นจึงต้องมีตัวเลขบอกว่าเรคอร์ดต่อไปเริ่มต้นที่ตำแหน่งใด

ข้อมูลของเวอร์กชีตจริง ๆ จะเริ่มเก็บตั้งแต่ไบต์ที่ 5 ขนาดความยาวของข้อมูลขึ้นกับชนิดของข้อมูล

อนึ่งตัวเลขบอกความยาวเรคอร์ดเป็นการบอกขนาดของส่วนของข้อมูลที่มีความยาวไม่เท่ากันในแต่ละ เรคอร์ด ส่วนของเรคอร์ดเฮดเดอร์มีความยาวคงที่คือ 4 ไบต์

เนื่องจากโลตัส 1-2-3 เขียนขึ้นเพื่อใช้กับซีพียู 8086/88 ซึ่งมีการอ่านตัวเลขในลักษณะอินเทลฟอร์แมต ดังนั้นขนาดตัวเลข 2 ไบต์จะมีไบต์แรกเป็นไบต์ที่มีนัยสำคัญต่ำกว่า เช่นถ้าจะบอกตัวเลข 06 ก็จะเก็บ 0600 เป็นต้น

ตัวอย่างเรคอร์ดที่ใช้ในโลตัส 1-2-3

เรคอร์ด BOF

เป็นเรคอร์ดบอกการเริ่มต้นของไฟล์โลตัส 1-2-3 ทำหน้าที่บอกว่าไฟล์นี้เป็นไฟล์โลตัส 1-2-3 ในรูปแบบ .WKS หรือ .WK1 โครงสร้างของข้อมูลเรคอร์ดนี้เป็นดังรูปที่ 3

รหัส 0000 ของเฮดเดอร์ มีความหมายว่าเป็นจุดเริ่มต้นของไฟล์ รหัสนี้จะบอกว่าเรคอร์ดนี้เป็นเรคอร์ด BOF จะมีข้อมูลบอกเวอร์ชันในส่วนของข้อมูล

ความยาวของเรคอร์ดเก็บ 0200 หมายถึง 2 ไบต์ ส่วนข้อมูลตัวเลข 0404 ให้ความหมายว่าเป็นโลตัส 1-2-3 เวอร์ชัน 1A

เรคอร์ดแสดงเซลล์พื้นฐาน

ในการแสดงข้อมูลในเซลล์นั้น โลตัสมีการบอกรายละเอียดของตำแหน่งเซลล์ เช่น เป็นแถวและคอลัมน์ เช่น ตัวอย่างในรูปที่ 4 เป็นตัวอย่างของการแสดงว่า cell F11 ไม่มีข้อมูลอะไรอยู่เป็นเซลล์ว่าง

รหัส 0C เป็นรหัสบอกว่าเซลล์นี้เป็น blank cell ส่วนของความยาวบอกว่ามีข้อมูลขนาด 5 ไบต์ ซึ่งไบต์แรกบอกฟอร์แมตของเซลล์นี้ ส่วนอีก 4 ไบต์ถัดไปบอกตำแหน่งแถวและคอลัมน์ ส่วนของฟอร์แมตเซลล์ในไบต์แรกของส่วนข้อมูลนี้เป็นลักษณะของการจัดการกำหนดรูปแบบการแสดงผลที่ผู้ใช้สร้างขึ้น จากตัวอย่างนี้รหัส 22 หมายถึง unprotected cell ที่มีการแสดงผลแบบ currency แสดงทศนิยม 2 ตำแหน่ง

ตัวอย่างของเซลล์อยู่ที่คอลัมน์ 05 หรือคอลัมน์ F และแถว 0F คือ แถวที่ 11 หรือเซลล์ F11 การบอกตำแหน่งนี้เราจะเริ่มที่ 00 ถ้าเป็นคอลัมน์ก็หมายถึงคอลัมน์ A และถ้าเป็นแถวก็จะหมายถึงแถวที่ 1

เซลล์เก็บข้อมูลเลขจำนวนเต็ม

สมมติมีตัวเลข 1245 เก็บไว้ในเซลล์ A1 ข้อมูลของเรคอร์ดที่เก็บข้อมูลนี้จะเป็นดังรูปที่ 5

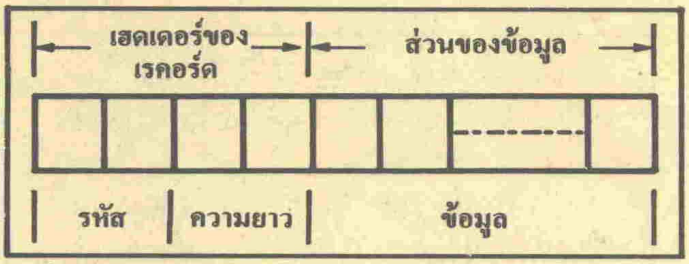
รหัส 0D เป็นรหัสบอกว่าเรคอร์ดนี้เป็นเรคอร์ดเก็บเลขจำนวนเต็ม ส่วนความยาวบอกว่ามีด้วยกันทั้งหมด 7 ไบต์ ส่วนฟอร์แมต 82 แสดงถึงการเป็น unprotected cell ที่แสดงผลเป็นตัวเลขมีจุดทศนิยมสองตำแหน่ง

การแสดงตำแหน่งเซลล์บอกว่าอยู่แถว 0 คอลัมน์ 0 ส่วนตัวเลข 04DD เป็นตัวเลขฐานสิบหกที่มีค่าเท่ากับ 1245 ในระบบตัวเลขฐานสิบ

เรคอร์ดเก็บเลเบล

ตัวอย่างในรูปที่ 6 นี้เป็นตัวอย่างข้อมูลที่เก็บข้อความ Label ที่เก็บในเซลล์แบบชิดซ้าย

รหัส 0F เป็นรหัสที่บอกว่าเซลล์นี้เป็นเลเบลหรือเก็บข้อความ ความยาว



รูปที่ 2 โครงสร้างเรคอร์ดในโลตัส

รูปที่ 3 โครงสร้างเรคอร์ด BOF

รูปที่ 4 โครงสร้างเรคอร์ดของเซลล์พื้นฐาน

12 ไบต์ (0C) เป็นการบอกว่าส่วนข้อมูลมี 12 ไบต์ (ไบต์ 0-11) ส่วนฟอร์แมตเซลล์เป็น FF หมายถึง default format ในกรณีนี้ global format จะนำมาใช้กับเซลล์นี้แทน

ส่วนของเลเบลจะต้องเริ่มตัวอักษรที่จะบอกรายละเอียดบางอย่าง ซึ่งตัวอักษรที่ใช้มีดังรูปที่ 7

ดังนั้นในตัวอย่างนี้จึงให้จัดชิดด้วย และเลเบลทุกตัว มีรหัส 00 (null) ปิดท้ายเรคอร์ดเสมอ

เรคอร์ดแสดงตัวเลขจำนวนจริง (real number)

ตัวอย่างที่แสดงในรูปที่ 8 เป็นการเก็บตัวเลข 10.1 ในเซลล์ D2

รหัส 0E เป็นรหัสบอกว่าเรคอร์ดที่เป็นเรคอร์ดเก็บเลขจำนวนจริง ตัวเลข 0D บอกจำนวน 13 ไบต์

รหัส FF บอกว่าเป็น protected และมีฟอร์แมตแบบ default

ตัวเลข 0300 บอกคอลัมน์ D

ตัวเลข 0100 บอกแถว 2

ส่วนไบต์ 5 ถึงไบต์ 12 เก็บตัวเลข float ขนาด 8 ไบต์ เป็นไปตามมาตรฐานของ IEEE

หากจะคำนวณหาค่าตัวเลขนี้จะต้องเรียงตามนัยสำคัญใหม่เป็น

40 24 33 33 33 33 33 33

ไบต์แรกเป็น 40 บิตแรก S=0 จึงหมายถึงเลขบวก

ค่าตัวเลข Exponent มีขนาด 11 บิต คือ 402h เมื่อแปลงเป็นเลขฐานสิบจะได้ผล 1026_d

ส่วนค่าแมนติสซ่าคือ 4 33 33 33 33 33 33 เมื่อคำนวณออกมาจะได้ 0.2625_d สังเกตค่า Exponent จะอยู่ระหว่าง 0 และ 2047 ดังนั้นค่าเลขจำนวนจริงในกรณีนี้หาได้ดังรูปที่ 9

ไบต์ที่	เฮดเดอร์				ข้อมูล						
	0	1	2	3	0	1	2	3	4	5	6
ตัวเลขฐานสิบหก	0D	00	07	00	82	00	00	00	00	DD	04
	รหัส		ความยาว		ฟอร์แมตเซลล์	หมายเลขคอลัมน์		หมายเลขแถว		ตัวเลขจำนวนเต็ม	

รูปที่ 5 โครงสร้างเซลล์เก็บข้อมูลเลขจำนวนเต็ม

ไบต์ที่	เฮดเดอร์				ข้อมูล											
	0	1	2	3	0	1	2	3	4	5	6	7	8	9	10	11
ตัวเลขฐานสิบหก	0F	00	0C	00	FF	00	00	00	00	2C	4C	61	62	65	6C	00
	รหัส		ความยาว		ฟอร์แมตเซลล์	หมายเลขคอลัมน์		หมายเลขแถว		Label						null

รูปที่ 6 โครงสร้างเรคอร์ดเก็บเลเบล

' รหัส ASCII 2Ch (ตัวเลข 44 ฐานสิบ) หมายถึง ชิดซ้าย
" รหัส ASCII 22h (ตัวเลข 34 ฐานสิบ) หมายถึง ชิดขวา
^ รหัส ASCII 5Eh (ตัวเลข 94 ฐานสิบ) หมายถึง จัดกึ่งกลาง
\ รหัส ASCII 5Ch (ตัวเลข 92 ฐานสิบ) หมายถึง การซ้ำ
| รหัส ASCII 7Ch (ตัวเลข 124 ฐานสิบ) หมายถึง คำสั่งควบคุมการพิมพ์

รูปที่ 7 อักษรบอกรายละเอียดของเลเบล

```
ตัวเลข = (-1)^S*2^(E-1023d)*(1.F)
       = (-1)^0*2^(1026d-1023d)*(1.2625)
       = 1*2^3*1.2625
       = 8*1.2625
       = 10.1
```

รูปที่ 9 การแปลงเลขจำนวนจริงขนาด 8 ไบต์ตามมาตรฐาน IEEE

เฮดเดอร์		ข้อมูล												
		0	1	2	3	4	5	6	7	8	9	10	11	12
OE	00	0D	00	FF	03	00	01	00	33	33	33	33	33	33	24	40

รูปที่ 8 โครงสร้างเรคอร์ดแสดงตัวเลขจำนวนจริง

เรคอร์ดเก็บสูตร

ตัวอย่างในรูปที่ 10 นี้เป็นตัวอย่างของโลตัสเก็บสูตร +F8*(17/@SIN(D2)) ในเซลล์ A1

ส่วนของเฮดเดอร์เก็บรหัส 10 บอกว่าเป็นเรคอร์ดเก็บสูตร ตัวเลข 21h บอกความยาวของข้อมูล 33 ไบต์

ความหมายของส่วนข้อมูลเป็นดังนี้

ไบต์ 0	FF	หมายถึง protected เป็นดีฟอลต์ฟอร์แมต
" 1-2	00 00	หมายถึง คอลัมน์ A
" 3-4	00 00	" แถว 1
" 5-12	เป็นตัวเลขที่แสดงแบบ real เป็นผลลัพธ์ของสูตร	
" 13-14	0012h = 18d	หมายถึงขนาดของสูตร 18 ไบต์
" 15-32	สตริงเก็บสูตร	

สตริงเก็บสูตรจะได้รับการจัดรูปแบบให้อยู่ในลักษณะ RPN (Reverse (Polish Notation) ลองพิจารณาจากสูตรในรูปที่ 11 ไบต์แรกคือไบต์ที่ 0 ของสูตรเก็บรหัส 01 หมายถึงตัวแปรซึ่งจะต้องตามมาอีก 4 ไบต์ที่จะบอกแถวและคอลัมน์ ในที่นี้ไบต์ที่ 1-2 บอกคอลัมน์ซึ่งมีค่า 0580 เมื่อเขียนให้ถูกหลักการของนัยสำคัญจะได้ 80 05 และ 2 ไบต์ถัดมาบอกแถวคือ 80 07 การบอกนี้เรามีบิตแรก (MSB) เป็น 1 บอกว่าตัวเลขแถวและคอลัมน์นี้เป็นแบบ relative หรือบอกว่าเซลล์ที่อ้างอิงนี้อยู่ห่างจากเซลล์ที่เก็บสูตรนี้เท่าไร จากกรณีนี้สูตรเก็บอยู่ที่ A1 การใช้ตัวแปรนี้อยู่ห่างจาก A1 ไปทางขวา 5 และลงมาข้างล่าง 7 หรือคือ F8 นั่นเอง ส่วนไบต์ที่ 5 เก็บตัวเลข 05h เป็นออปโค้ด มีความหมายถึงเป็นตัวเลขจำนวนเต็มซึ่งในที่นี้เก็บในสองไบต์ถัดไปคือ 11 00 เมื่อจับเรียงใหม่ได้ 0011h หรือมีค่าเท่ากับ 17_d นั่นเอง

ไบต์ที่ 8 เก็บตัวเลข 01 เป็นการบอกว่าเป็นตัวแปร ตัวเลข 0380, 01 80 จะบอกตำแหน่งเซลล์ซึ่งก็คือ D2 (ตามหลักการที่กล่าวแล้ว)

ไบต์ที่ 13 เก็บค่าตัวเลข 27h ซึ่งเป็นรหัสของ @SIN(X)

ไบต์ที่ 14 เก็บค่าตัวเลข 0Ch เป็นรหัสของตัวโอเปอเรเตอร์หาร

ไบต์ที่ 15 เก็บ 04h หมายถึงวงเล็บ

ไบต์ที่ 16 เก็บ 0Bh เป็นรหัสตัวโอเปอเรเตอร์ คูณ

ไบต์ที่ 17 เก็บรหัสจบสูตรซึ่งใช้ออปโค้ด 03

เมื่อเขียนในฟอร์ม RPN จะได้

F8, 17, D2, @SIN (X), /, (,*

ในการคำนวณตามสูตรนี้ โลตัสจะอ่านสตริงจากซ้ายไปขวาแล้วพุชค่าตัวเลขลงไปในสแต็ก แล้วกระทำตามโอเปอเรเตอร์ทันที

การจัดเรคอร์ดเป็นไฟล์

ในการเก็บข้อมูลลงไฟล์ WKS หรือ WK1 โลตัสจัดโครงสร้างเรคอร์ดเรียงกันไปในไฟล์เรียงลำดับกันไป โดยแต่ละเรคอร์ดมีความยาวไม่เท่ากันขึ้นกับชนิดของเรคอร์ดนั้น ๆ

โครงสร้างของไฟล์ WKS หรือ WK1 มีดังนี้

BOF

WKSPASS (รหัส password ใช้ในเวอร์ชัน 2 เท่านั้น)

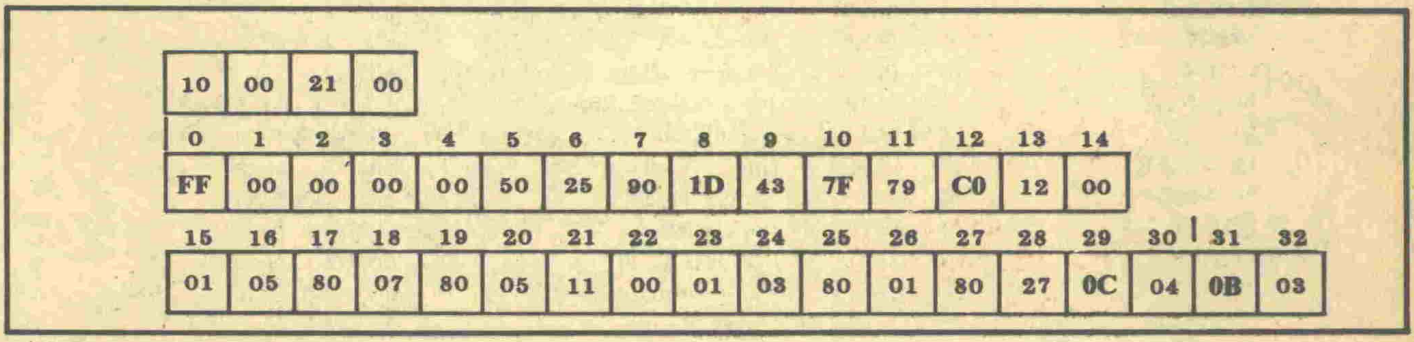
DIMENSIONS

CPI

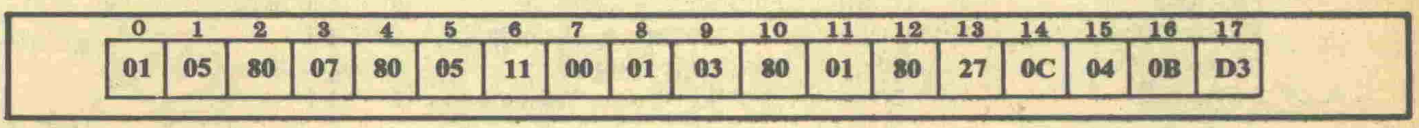
CALCCOUNT

CALCMODE

CALORDER



รูปที่ 10 โครงสร้างเรคอร์ดเก็บสูตร



รูปที่ 11

SPLIT
SYNC
Window 1 Descriptor
    WINDOW 1
    COLW 1
    HIDCOL 1 (มีในเวอร์ชัน 2 เท่านั้น)
Window 2 Descriptor
    WINDOW 2
    COLW 2
    HIDCOL 2 (มีในเวอร์ชัน 2 เท่านั้น)
CURSORW/2
NRANGE
NGRAPH
TABLE
QRANGE
PRANGF

FORMAT
FRANGE
SRANGE
KRANGE 1
KRANGE 2 (มีในเวอร์ชัน 2 เท่านั้น)
MRANGES (มีในเวอร์ชัน 2 เท่านั้น)
DRANGE
PARSE
PROTECT
FOOTER
HEADER
SETUP
MARGINS
LABELFMT
TITLES
GRAPH

Cell list:
    BLANK
    or INTEGER
    or NUMBER
    or LABEL
    or FORMULA
    or FORMULA and STRING (มีใน 123 เวอร์ชัน 2 เท่านั้น)

สังเกตว่าชนิดของเรคอร์ดที่บอกรายละเอียดเกี่ยวกับเวอร์กชีตมีมากตามรหัสที่กำหนด โลตัสได้วางรหัสไว้โดยรหัสแต่ละตัวจะมีความหมายเฉพาะ โดยรายละเอียดของเรคอร์ดเหล่านี้หาดูได้จากหนังสือ Lotus file formats for 123 Symphony & Jazz ซึ่งทางบริษัทโลตัสได้จัดพิมพ์ขึ้น

ในที่นี้ได้นำตารางรหัสของชนิดเรคอร์ดมาแสดงดังตารางที่ 1

ตารางที่ 1 แสดงรหัสความหมายเรคอร์ดที่ใช้ในโลตัส

ชื่อเรคอร์ด	รหัส		ความหมาย	ใช้ในซอฟต์แวร์
BOF	0	[00h]	Beginning of File	all 1-2-3, Sym
EOF	1	[01h]	End of File	all 1-2-3, Sym
CALCMODE	2	[02h]	Calculation Mode	all 1-2-3, Sym
CALCORDER	3	[03h]	Calculation Order	all 1-2-3, Sym
SPLIT	4	[04h]	Split Window Mode	all 1-2-3
SYNC	5	[05h]	Split Window Sync	all 1-2-3
DIMENSIONS	6	[06h]	Worksheet Dimensions	all 1-2-3, Sym
WINDOW1	7	[07h]	Window 1	all 1-2-3
COLW1	8	[08h]	Column Width, Window 1	all 1-2-3
WINDOW2	9	[09h]	Window 2	all 1-2-3
COLW2	10	[0Ah]	Column Width, Window 2	all 1-2-3
NRANGE	11	[0Bh]	1-2-3 Named Range	1-2-3/1, 1-2-3/1A
BLANK	12	[0Ch]	Blank Cell	all 1-2-3, Sym
INTEGER	13	[0Dh]	Integer Cell	all 1-2-3, Sym
NUMBER	14	[0Eh]	Number Cell	all 1-2-3, Sym
LABEL	15	[0Fh]	Label Cell	all 1-2-3, Sym
FORMULA	16	[10h]	Formula Cell	all 1-2-3, Sym
TABLE	24	[18h]	Table Range	all 1-2-3, Sym
QRANGE	25	[19h]	Query Range	all 1-2-3, Sym/1.1
PRANGE	26	[1Ah]	Print Range	all 1-2-3, Sym/1.1
SRANGE	27	[1Bh]	Sort Range	all 1-2-3
FRANGE	28	[1Ch]	Fill Range	all 1-2-3, Sym
KRANGE1	29	[1Dh]	Sort Key Range, Primary	all 1-2-3
DRANGE	32	[20h]	Distribution Range	all 1-2-3, Sym
KRANGE2	35	[23h]	Sort Key Range, Secondary	all 1-2-3

▶

ตารางที่ 1 (ต่อ)

ชื่อเรคอร์ด	รหัส		ความหมาย	ใช้ในซอฟต์แวร์
PROTECT	36	[24h]	Global Protection	all 1-2-3, Sym
FOOTER	37	[25h]	Print Footer	all 1-2-3
HEADER	38	[26h]	Print Header	all 1-2-3
SETUP	39	[27h]	Print Setup	all 1-2-3
MARGINS	40	[28h]	Print Margins	all 1-2-3
LABELFMT	41	[29h]	Global Label Alignment	all 1-2-3, Sym
TITLES	42	[2Ah]	Print Borders	all 1-2-3
GRAPH	45	[2Dh]	1-2-3 Graph Settings	all 1-2-3
NGRAPH	46	[2Eh]	1-2-3 Named Graph Settings	all 1-2-3
CALCCOUNT	47	[2Fh]	Iteration Count	all 1-2-3, Sym
FORMAT	48	[30h]	Print Format Mode	all 1-2-3
CURSORW12	49	[31h]	Cursor in Window 1 or 2	all 1-2-3
BOF	0	[00h]	Beginning of File	all 1-2-3, Sym
EOF	1	[01h]	End of File	all 1-2-3, Sym
CALCMODE	2	[02h]	Calculation Mode	all 1-2-3, Sym
CALCORDER	3	[03h]	Calculation Order	all 1-2-3, Sym
SPLIT	4	[04h]	Split Window Mode	all 1-2-3
SYNC	5	[05h]	Split Window Sync	all 1-2-3
DIMENSIONS	6	[06h]	Worksheet Dimensions	all 1-2-3, Sym
WINDOW1	7	[07h]	Window 1	all 1-2-3
COLW1	8	[08h]	Column Width, Window 1	all 1-2-3
WINDOW2	9	[09h]	Window 2	all 1-2-3
COLW2	10	[0Ah]	Column Width, Window 2	all 1-2-3
BLANK	12	[0Ch]	Blank Cell	all 1-2-3, Sym
INTEGER	13	[0Dh]	Integer Cell	all 1-2-3, Sym
NUMBER	14	[0Eh]	Number Cell	all 1-2-3, Sym
LABEL	15	[0Fh]	Label Cell	all 1-2-3, Sym
FORMULA	16	[10h]	Formula Cell	all 1-2-3, Sym
TABLE	24	[18h]	Table Range	all 1-2-3, Sym
QRANGE	25	[19h]	Query Range	all 1-2-3
PRANGE	26	[1Ah]	Print Range	all 1-2-3
SRANGE	27	[1Bh]	Sort Range	all 1-2-3
FRANGE	28	[1Ch]	Fill Range	all 1-2-3, Sym
KRANGE1	29	[1Dh]	Sort Key Range, Primary	all 1-2-3
DRANGE	32	[20h]	Distribution Range	all 1-2-3, Sym
KRANGE2	35	[23h]	Sort Key Range, Secondary	all 1-2-3
PROTECT	36	[24h]	Global Protection	all 1-2-3, Sym
FOOTER	37	[25h]	Print Footer	all 1-2-3
HEADER	38	[26h]	Print Header	all 1-2-3
SETUP	39	[27h]	Print Setup	all 1-2-3
MARGINS	40	[28h]	Print Margins	all 1-2-3
LABELEMT	41	[29h]	Global Label Alignment	all 1-2-3, Sym
TITLES	42	[2Ah]	Print Borders	all 1-2-3
GRAPH	45	[2Dh]	1-2-3 Graph Settings	all 1-2-3
NGRAPH	46	[2Eh]	1-2-3 Named Graph Settings	all 1-2-3
CALCCOUNT	47	[2Fh]	Iteration Count	all 1-2-3, Sym
FORMAT	48	[30h]	Print Format Mode	all 1-2-3
CURSORW12	49	[31h]	Cursor in Window 1 or 2	all 1-2-3
STRING	51	[33h]	Formula String	1-2-3/2, all Sym
SNRANGE	71	[47h]	Symphony Named Range	1-2-3/2, all Sym
WKSPASS	75	[4Bh]	Worksheet Password	1-2-3/2, Sym/1.1
HIDCOL1	100	[64h]	Hidden Columns, Screen 1	1-2-3/2 ▶

ตารางที่ 1 (ต่อ)

ชื่อเรคอร์ด	รหัส	ความหมาย	ใช้ในซอฟต์แวร์
HIDCOL2	101 [65h]	Hidden Columns, Screen 2	1-2-3/2
PARSE	102 [66h]	Parse Ranges	1-2-3/2
RRANGES	103 [67h]	Linear Regression Ranges	1-2-3/2
MRANGES	105 [69h]	Matrix Math Ranges	1-2-3/2
CPI	150 [96h]	Cell Pointer Index	1-2-3/2, Sym/1.1

ตารางที่ 2 แสดงรหัสออปโค้ดของโลตัส

รหัส	ความหมาย
0 [0h]	ค่าคงตัว ตัวเลขจำนวนจริง 5 ไบต์ ตามมาตรฐาน IEEE
1 [1h]	ตัวแปร ตัวเลขตามมา 4 ไบต์ ไบต์ 0-1 ที่ตามมาแสดงคอลัมน์ ไบต์ 2-3 ที่ตามมาแสดงแถว
2 [2h]	พิสัย (range) รหัสที่ตามมาด้วยตัวเลข 8 ไบต์ ไบต์ 0-1 ที่ตามมาแสดงคอลัมน์เริ่มต้น ไบต์ 2-3 ที่ตามมาแสดงแถวเริ่มต้น ไบต์ 4,5 ที่ตามมาแสดงคอลัมน์สุดท้าย ไบต์ 6,7 ที่ตามมาแสดงแถวสุดท้าย
3 [3h]	รีเทิร์น เป็นรหัสบอกจบสูตร
4 [4h]	วงเล็บ จะไม่กระทำขณะการคำนวณ
5 [5h]	เลขคงตัวจำนวนเต็ม เก็บเลข 2 ไบต์
6 [6h]	ค่าคงตัวสตริง ความยาวตัวแปร ปิดท้ายด้วย NULL

รหัสที่ใช้ในสูตร

ส่วนของการกำหนดรหัสอีกส่วนหนึ่งคือรหัสที่ใช้ในสูตรที่โลตัสได้นิยามไว้เป็นรหัสดังแสดงในตารางที่ 2 เพื่อให้สูตรสามารถเขียนได้กะทัดรัดและจัดฟอร์มอยู่ในรูปของ RPN ได้โดยง่าย การจัดรหัสนี้จะทำให้การเขียนสูตรด้วยโลตัสทำไปอย่างมีระบบ

โลตัสยังได้จัดรหัสสำหรับตัวโอเปอเรเตอร์และฟังก์ชันสำหรับใช้งานต่าง ๆ เป็นรหัส โดยแยกเป็นกลุ่มที่ใช้ตัวพารามิเตอร์เพียงตัวเดียวและหลายตัว ตารางที่ 3 เป็นตารางแสดงรหัสของตัวโอเปอเรเตอร์ที่ใช้พารามิเตอร์ที่เป็นอาร์กิวเมนต์ 1 ตัว ส่วนตารางที่ 4 เป็นตารางที่ฟังก์ชันของโลตัสที่ใช้อาร์กิวเมนต์หลายตัวเป็นพารามิเตอร์ ส่วนตารางที่ 5 แสดงลำดับของโอเปอเรเตอร์ที่โลตัสได้จัดลำดับไว้

ตารางที่ 3 โอเปอเรเตอร์ที่มีอาร์กิวเมนต์ 1 ตัว

8	[8h]	unary, −	เลขลบ
9	[9h]	+	บวก
10	[Ah]	−	ลบ
11	[Bh]	*	คูณ
12	[Ch]	/	หาร
13	[Dh]	^	ยกกำลัง เช่น 3^2 = 3*3
14	[Eh]	=	เท่ากับ
15	[Fh]	< >	ไม่เท่ากับ
16	[10h]	<=	น้อยกว่าหรือเท่ากับ
17	[11h]	>=	มากกว่าหรือเท่ากับ
18	[12h]	<	น้อยกว่า
19	[13h]	>	มากกว่า
20	[14h]	#AND#	ลอจิก AND
21	[15h]	#OR#	ลอจิก OR
22	[16h]	# NOT #	ลอจิก NOT

23	[17h]	unary, +	(ignored during recalculation)
31	[1Fh]	@NA	
32	[20h]	@ERR	
33	[21h]	@ABS	
34	[22h]	@INT	
35	[23h]	@SQRT	
36	[24h]	@LOG	
37	[25h]	@LN	
38	[26h]	@PI	3.141592653589793
39	[27h]	@SIN	
40	[28h]	@COS	
41	[29h]	@TAN	
42	[2Ah]	@ATAN2	
43	[2Bh]	@ATAN	
44	[2Ch]	@ASIN	
45	[2Dh]	@ACOS	
46	[2Eh]	@EXP	
47	[2Fh]	@MOD	
49	[31h]	@ISNA	
50	[32h]	@ISERR	
51	[33h]	@FALSE	
52	[34h]	@TRUE	
53	[35h]	@RAND	
55	[37h]	@TODAY	1-2-3 เท่านั้น
		@NOW	ใช้เฉพาะใน Symphony
60	[3Ch]	@DAY	
61	[3Dh]	@MONTH	
63	[3Fh]	@ROUND	
65	[41h]	@HOUR	
66	[42h]	@MINUTE	
67	[43h]	@SECOND	
68	[44h]	@ISNUMBER	
69	[45h]	@ISSTRING	
70	[46h]	@LENGTH	
71	[47h]	@VALUE	
72	[48h]	@STRING	
74	[4Ah]	@CHAR	
75	[4Bh]	@CODE	
78	[4Eh]	@TIMEVALUE	
79	[4Fh]	@CELLPOINTER	

ตารางที่ 4 ตัวโอเปอเรเตอร์ที่ใช้อาร์กิวเมนต์หลายตัว

ฟังก์ชันนี้ใช้รหัสตามที่แสดงในตาราง และตามด้วยอาร์กิวเมนต์เป็นจำนวนไบต์ตามตัวนับบอกขนาดข้อมูล

48	[30h]	@CHOOSE
54	[36h]	@DATE
56	[38h]	@PMT
57	[39h]	@PV
58	[3Ah]	@FV

59	[3Bh]	@IF	
62	[3Eh]	@YEAR	
64	[40h]	@TIME	
73	[49h]	@MID	
76	[4Ch]	@FIND	
77	[4Dh]	@DATEVALUE	
80	[50h]	@SUM	
81	[51h]	@AVG	
82	[52h]	@COUNT	
83	[53h]	@MIN	
84	[54h]	@MAX	
85	[55h]	@VLOOKUP	
86	[56h]	@NPV	
87	[57h]	@VAR	
88	[58h]	@STD	
89	[59h]	@IRR	
90	[5Ah]	@HLOOKUP	
91	[5Bh]	@DSUM	
92	[5Ch]	@DAVG	
93	[5Dh]	@DCOUNT	
94	[5Eh]	@DMIN	
95	[5Fh]	@DMAX	
96	[60h]	@DVAR	
97	[61h]	@DSTD	
98	[62h]	@INDEX	
99	[63h]	@COLS	
100	[64h]	@ROWS	
101	[65h]	@REPEAT	
102	[66h]	@UPPER	
103	[67h]	@LOWER	
104	[68h]	@LEFT	
105	[69h]	@RIGHT	
106	[6Ah]	@REPLACE	
107	[6Bh]	@PROPER	
108	[6Ch]	@CELL	
109	[6Dh]	@TRIM	
110	[6Eh]	@CLEAN	
111	[6Fh]	@S	
112	[70h]	@N	
113	[71h]	@EXACT	
114	[72h]	@CALL(@APP)	; ใช้ในซิมโฟนี
115	[73h]	@@	
116	[74h]	@RATE	
117	[75h]	@TERM	
118	[76h]	@CTERM	
119	[77h]	@SLN	
120	[78h]	@SYD	; Sum of Years
121	[79h]	@DDB	Depreciation
156	[9Ch]	reserved	
206	[CEh]	reserved	
255	[FFh]	reserved	

ตารางที่ 5 ลำดับความสำคัญของการคำนวณ

โอเปอเรเตอร์	ตัวที่อยู่ก่อน แบบ unary	ตัวที่อยู่ก่อน แบบ binary
+	6	4
−	6	4
*	na	5
/	na	5
^	na	7 ;highest precedence
=	na	3
<>	na	3
<=	na	3
>=	na	3
<	na	3
>	na	3
#AND#	na	1 ;lowest precedence
#OR#	na	1 ;lowest precedence
#NOT#	2	na

ลองดูจากตัวอย่าง

ตัวอย่างต่อไปนี้เป็นการทดลองสร้างชิ้นงาน ซึ่งประกอบด้วยตัวเลขในคอลัมน์ x และ y และสูตร x+y โดยที่แถวสุดท้ายเป็นการ @SUM ของตัวเลข ลักษณะของชิ้นงานแสดงไว้ดังรูปที่ 12

ตัวเวอร์กชีตประกอบด้วยเลเบลใน A1, A2, B2, C2, A7 และ A9 ส่วนของ A3..B6 เป็นตัวเลขโดยที่ A3..A6 เป็นเลขจำนวนเต็ม C3..C6 และ A8..C8 เป็นสูตร สูตรใน C3 เก็บ A3+B3 สูตรใน A8 เก็บ @SUM (A3..A6)

เมื่อเก็บข้อมูลไว้จะได้ข้อมูลชื่อ SAMI.WKI ลองใช้ดีบักเรียกออกมาดูจะเห็นข้อมูลเป็นตัวเลขฐานสิบหก ขอให้สังเกตการจัดเรียงข้อมูลซึ่งในที่นี้แอดเดรส 100 ตามรูปที่ 13 คือจุดเริ่มต้นของข้อมูลในไฟล์

ข้อมูลส่วนต้นจากแอดเดรส 100 จะเรียงกันไปเป็นเรคอร์ดที่มีความหมายดังรูปที่ 14

```
A1: 'Sample Worksheet                                          READY

     A          B          C          D          E          F          G          H
1    Sample Worksheet
2              x          y        x+y
3              1        2.3        3.3
4              4        4.5        8.5
5              7        6.7       13.7
6             11        8.1       19.1
7    ============================
8       23.00      21.60      44.60
9    ============================
10
11
12
13
14
15
16
17
18
19
20
01-Jan-80  12:03 AM
```

รูปที่ 12 เวอร์กชีตตัวอย่าง

```
-d
0918:0100  00 00 02 00 06 04 06 00-08 00 00 00 00 00 02 00   ................
0918:0110  08 00 96 00 12 00 00 00-00 00 08 00 01 00 01 00   ................
0918:0120  07 00 02 00 01 00 07 00-2F 00 01 00 01 02 00 01   ......../.......
0918:0130  00 FF 03 00 01 00 00 04-00 01 00 00 05 00 01 00   ................
0918:0140  FF 07 00 20 00 00 00 00-02 00 71 00 09 00 08 00   ... ......q.....
0918:0150  00 00 00 00 00 00 00 00-00 00 00 00 00 04 00 04   ................
0918:0160  00 48 00 00 00 64 00 20-00 00 00 00 00 00 00 00   .H...d. ........
0918:0170  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-d
0918:0180  00 00 00 00 00 00 00 00-00 18 00 19 00 00 FF FF   ................
0918:0190  00 00 FF FF 00 00 FF FF-00 00 FF FF 00 00 FF FF   ................
0918:01A0  00 00 FF FF 00 00 19 00-19 00 FF FF 00 00 FF FF   ................
0918:01B0  00 00 FF FF 00 00 FF FF-00 00 FF FF 00 00 FF FF   ................
0918:01C0  00 00 00 1A 00 08 00 FF-FF 00 00 FF FF 00 00 30   ...............0
0918:01D0  00 01 00 00 1C 00 08 00-FF FF 00 00 FF FF 00 00   ................
0918:01E0  1B 00 08 00 FF FF 00 00-FF FF 00 00 1D 00 09 00   ................
0918:01F0  FF FF 00 00 FF FF 00 00-00 23 00 09 00 FF FF 00   .........#......
-d
0918:0200  00 FF FF 00 00 00 67 00-19 00 FF FF 00 00 FF FF   ......g.........
0918:0210  00 00 FF FF 00 00 FF FF-00 00 FF FF 00 00 FF FF   ................
0918:0220  00 00 00 69 00 28 00 FF-FF 00 00 FF FF 00 00 FF   ...i.(..........
0918:0230  FF 00 00 FF FF 00 00 FF-FF 00 00 FF FF 00 00 FF   ................
0918:0240  FF 00 00 FF FF 00 00 FF-FF 00 00 FF FF 00 00 20   ...............
0918:0250  00 10 00 FF FF 00 00 FF-FF 00 00 FF FF 00 00 FF   ................
0918:0260  FF 00 00 66 00 10 00 FF-FF 00 00 FF FF 00 00 FF   ...f............
0918:0270  FF 00 00 FF FF 00 00 24-00 01 00 00 25 00 F2 00   .......$....%.r.
-d
```

รูปที่ 13 ข้อมูลในไฟล์เวอร์กชีตตัวอย่าง (รูปที่ 12)

รูปที่ 14 รายละเอียดของแต่ละเรคอร์ดตั้งแต่แอดเดรส 100 ลงไป

BOF	- 00 00 02 00 06 04
WKSPASS	- ในที่นี้ไม่มีการตั้งรหัสผ่าน
DIMENSIONS	- 06 00 08 00 00 00 00 00 02 00 08 00 หมายถึงใช้เนื้อที่ 3 คอลัมน์ 9 แถว
CPI	- 96 00 12 00 ... ตามด้วยข้อมูล 18 ไบต์
CALCCOUNT	- 2F 00 01 00 01
CALCMODE	- 02 00 01 00 FF
CALCORDER	- 03 00 01 00 00
SPLIT	- 04 00 01 00 00
SYNC	- 05 00 01 00 FF
WINDOW 1	- 07 00 20 00 นับต่อไปอีก 32 ไบต์
SGRAPH	- 64 00 20 00 นับต่อไปอีก 32 ไบต์
TABLE	- 18 00 19 00 นับต่อไปอีก 25 ไบต์
QRANGE	- 19 00 19 00 นับต่อไปอีก 25 ไบต์
PRANGE	- 1A 00 08 00 FF FF 00 00 FF FF 00 00
FORMAT	- 30 00 01 00 00
FRANGE	- 1C 00 08 00 FF FF 00 00 FF FF 00 00
SRANGE	- 1B 00 08 00 FF FF 00 00 FF FF 00 00
KRANGE 1	- 1D 00 09 00 FF FF 00 00 FF FF 00 00 00
KRANGE 2	- 23 00 09 00 นับไป 9 ไบต์
RRANGES	- 67 00 19 00 นับไป 25 ไบต์เป็นข้อมูล
MRANGES	- 69 00 28 00 นับไป 40 ไบต์
DRANGES	- 20 00 10 00 นับต่อไป 16 ไบต์เป็นข้อมูล

ข้อมูลต่อไปเป็นเซตกราฟ ถ้าไม่มีการเซตกราฟส่วนนี้จะแตกต่างออกไปจนถึงแอดเดรส 676 ซึ่งดูจากการแสดงจะเห็นข้อมูลในแต่ละเซลล์ปรากฏอยู่ดังรูปที่ 15

ขอให้ลองแยกเรคอร์ดของเลเบลตัวเลขและสูตรออกจากข้อมูลตั้งแต่แอดเดรส 676 ถึง 851 ซึ่งแอดเดรส 4 ไบต์สุดท้ายเป็นเรคอร์ด EOF

รูปที่ 14 (ต่อ)

PARSE	- 66	00	10	00		นับต่อไป 16 ไบต์
PROTECT	- 24	00	01	00	00	
FOOTER	- 25	00	F2	00		นับต่อไป 242 ไบต์
HEADER	- 26	00	F2	00		นับต่อไปอีก 242 ไบต์
SETUP	- 27	00	28	00		นับต่อไปอีก 40 ไบต์
MARGINS	- 28	00	0A	00		นับต่อไปอีก 10 ไบต์
LABEL	- 29	00	01	00	27	
TITLE	- 2A	00	10	00		นับต่อไปอีก 10 ไบต์

```
0918:0670  71 71 01 00 00 00 0F 00-17 00 FF 00 00 00 00 27   qq.............
-d                                  ↑เริ่มต้น Cell list
0918:0680  53 61 6D 70 6C 65 20 57-6F 72 6B 73 68 65 65 74   Sample Worksheet
0918:0690  00 0F 00 08 00 FF 00 00-01 00 22 78 00 0F 00 08   ..........."x....
0918:06A0  00 FF 01 00 01 00 22 79-00 0F 00 0A 00 FF 02 00   ......"y........
0918:06B0  01 00 22 78 2B 79 00 0D-00 07 00 FF 00 00 02 00   .."x+y..........
0918:06C0  01 00 0E 00 0D 00 FF 01-00 02 00 66 66 66 66 66   ...........fffff
0918:06D0  66 02 40 10 00 1B 00 FF-02 00 02 00 66 66 66 66   f.@.........ffff
0918:06E0  66 66 0A 40 0C 00 01 FE-BF 00 80 01 FF BF 00 80   ff.@...~?....?..
0918:06F0  09 03 0D 00 07 00 FF 00-00 03 00 04 00 0E 00 0D   ................
-d
0918:0700  00 FF 01 00 03 00 00 00-00 00 00 00 12 40 10 00   .............@..
0918:0710  1B 00 FF 02 00 03 00 00-00 00 00 00 00 21 40 0C   .............!@.
0918:0720  00 01 FE BF 00 80 01 FF-BF 00 80 09 03 0D 00 07   ..~?....?.......
0918:0730  00 FF 00 00 04 00 07 00-0E 00 0D 00 FF 01 00 04   ................
0918:0740  00 CD CC CC CC CC CC 1A-40 10 00 1B 00 FF 02 00   .MLLLLL.@.......
0918:0750  04 00 67 66 66 66 66 66-2B 40 0C 00 01 FE BF 00   ..gfffff+@...~?.
0918:0760  80 01 FF BF 00 80 09 03-0D 00 07 00 FF 00 00 05   ...?............
0918:0770  00 0B 00 0E 00 0D 00 FF-01 00 05 00 33 33 33 33   ............3333
-d
0918:0780  33 33 20 40 10 00 1B 00-FF 02 00 05 00 9A 99 99   33 @............
0918:0790  99 99 19 33 40 0C 00 01-FE BF 00 80 01 FF BF 00   ...3@...~?....?.
0918:07A0  80 09 03 0F 00 23 00 FF-00 00 06 00 27 3D 3D 3D   .....#......'===
0918:07B0  3D 3D 3D 3D 3D 3D 3D 3D-3D 3D 3D 3D 3D 3D 3D 3D   ================
0918:07C0  3D 3D 3D 3D 3D 3D 3D 3D-3D 00 10 00 1B 00 82 00   =========.......
0918:07D0  00 07 00 00 00 00 00 00-00 37 40 0C 00 02 00 80   .........7@.....
0918:07E0  FB BF 00 80 FE BF 50 01-03 10 00 1B 00 82 01 00   {?..~?P.........
0918:07F0  07 00 9A 99 99 99 99 99-35 40 0C 00 02 00 80 FB   ........5@.....{
-d
0918:0800  BF 00 80 FE BF 50 01 03-10 00 1B 00 82 02 00 07   ?..~?P..........
0918:0810  00 CD CC CC CC CC 4C 46-40 0C 00 02 00 80 FB BF   .MLLLLLF@.....{?
0918:0820  00 80 FE BF 50 01 03 0F-00 23 00 FF 00 00 08 00   ..~?P....#......
0918:0830  27 3D 3D 3D 3D 3D 3D 3D-3D 3D 3D 3D 3D 3D 3D 3D   '===============
0918:0840  3D 3D 3D 3D 3D 3D 3D 3D-3D 3D 3D 3D 3D 00 01 00   =============...
0918:0850  00 00 FA 75 2E 8B F7 E8-3D FA 75 27 8B DA EB 08   ..zu..wh=zu'.Zk.
0918:0860  90 E8 3C F4 75 1D 8B DA-49 83 F9 00 76 03 E9 39   .h<tu..ZI.y.v.i9
0918:0870  FF 8B 36 1E 00 E8 1F FA-75 09 B0 0D E8 03 FE 75   ..6..h.zu.0.h.~u
                                                        EOF
```

รูปที่ 15 ข้อมูลในไฟล์เวอร์กชีตตัวอย่าง (ส่วนท้าย)

อนึ่งในการเขียนโปรแกรมเพื่อแปลงข้อมูลระหว่างไฟล์อื่นมาเป็นโลตัส หรือจากโลตัสไปเป็นไฟล์อื่น เราจะต้องสร้างเรคอร์ดในเซลล์ต่าง ๆ ดังที่แสดงมานี้

ส่งท้าย

ข้อมูลที่เขียนขึ้นมานี้มีจุดประสงค์ที่จะให้ผู้อ่านสามารถเข้าใจหลักการเก็บข้อมูลในไฟล์ WKl ได้ อย่างไรก็ดี หากสนใจและต้องการรายละเอียดเพิ่มเติมขอให้ค้นจากหนังสือ Lotus file formats for 123 ของสำนักพิมพ์ Addison-Wesley 1986.